

دانشگاه تهران
دانشکده‌ی مهندسی برق و کامپیوتر



سیستم‌های نهفته‌ی بی‌درنگ

تمرین اول

سامانه نهفته آبیاری هوشمند با استفاده از بردهای Arduino

نیمسال دوم ۱۴۰۳-۴

سامانه نهفته آبیاری هوشمند با استفاده از بردهای Arduino

هدف

هدف این تمرین طراحی و شبیه‌سازی یک سامانه آبیاری هوشمند با استفاده از بردهای Arduino در نرم‌افزار Proteus است. این سامانه باید قابلیت پایش رطوبت خاک، دمای محیط، آبیاری خودکار و چرخاندن گلدان‌ها برای دریافت نور بیشتر را نشان دهد.

مشخصات سامانه

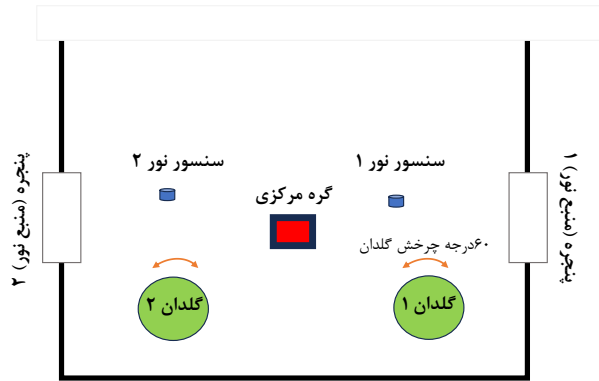
این سامانه آبیاری هوشمند شامل دو گلدان مجهز به سنسور و شیر آب اتوماتیک است که توسط یک کنترل کننده مرکزی کنترل می‌شوند. بخشهای دیجیتال این سامانه شامل بخش‌های زیر است:

۱. دو گره لبه: (Edge Node)

- هر گره لبه، سطح رطوبت خاک را در یک گلدان پایش می‌کند و کنترل می‌کند و شامل عملگرهای زیر است:
 - یک موتور برای باز و بسته کردن شیر آب.
 - یک موتور سروو برای چرخاندن گلدان به سمت نور.
- هر گره لبه یک برد Arduino دارد و از طریق پروتکل I2C با گره مرکزی ارتباط برقرار می‌کند.

۲. گره مرکزی: (Central Node)

- گره مرکزی دو وظیفه دارد:
 - گره‌های لبه را برای آبیاری بر اساس داده‌های رطوبت و دما مدیریت می‌کند.
 - دمای محیط را با استفاده از یک سنسور دما و رطوبت خاک را از گره لبه در هر گلدان دریافت می‌کند. این گره تصمیم می‌گیرد که بر اساس شرایط، آبیاری قطره‌ای صورت بگیرد یا خیر و اگر جواب مثبت است، این کار را با چه نرخ انجام دهد. برنامه‌ی باید تحت شرایط زیر، خروجی‌های متناسب را تولید و جهت اجرا به برد گره در گلدانها انتقال دهد: (۱) اگر رطوبت بالای ۸۰ درصد بود، آبیاری صورت نگیرد. (۲) اگر رطوبت کم‌تر از ۵۰ درصد بود، آبیاری با نرخ ۱۵ سی‌سی بر دقیقه انجام گیرد. (۳) اگر رطوبت بین ۵۰ تا ۸۰ درصد بود نیز دو حالت رخ می‌دهد. اگر دما بیش‌تر از ۲۵ درجه‌ی سلسیوس بود، آبیاری با نرخ ۱۰ قطره بر دقیقه صورت گیرد. اگر دما کم‌تر از ۲۵ درجه‌ی سلسیوس بود، آبیاری با نرخ ۵ قطره بر دقیقه صورت گیرد.
- شدت نور را با استفاده از دو سنسور نور پایش می‌کند و بر اساس مقایسه شدت نور از دو سنسور، گلدان‌ها را به سمت منبع نوری که شدت بیشتری دارد می‌چرخاند. گلدانها می‌توانند در دو موقعیت با فاصله ۶۰ درجه اختلاف قرار گیرند (شکل ۱) که بسته به خروجی سنسور نور در موقعیتی که نور بیشتر است قرار خواهند گرفت.



شکل ۱- موقعیت منبع نور و جهت گردش گلدان ها

قطعه های مورد نیاز

بخش پردازی:

- سه عدد برد Arduino Uno که دو تا در گلدانها و یکی در گره مرکزی قرار دارد و با رابط I2C به هم متصل هستند

سنسورها:

۱. گره های لبه:

- **Soil Moisture Sensor:** سطح رطوبت خاک را پایش کرده و داده ها را به گره مرکزی ارسال می کند.

- نوع سنسور پیشنهادی: آنالوگ مانند FC-28

۲. گره مرکزی:

- سنسور دما:

- آنالوگ مانند LM35

- دیجیتال (اختیاری) مانند DHT11/DHT22 :

Light Sensors: ○

- آنالوگ LDR: همراه با مدار تقسیم ولتاژ.

- دیجیتال (اختیاری): سنسور نوری مبتنی بر I2C مانند BH1750

عملگرها: (Actuators)

۱. موتور آبیاری:

- یک موتور DC که توسط گره لبه کنترل می شود تا شیر آب را باز و بسته کند.

۲. موتور چرخش گلدان:

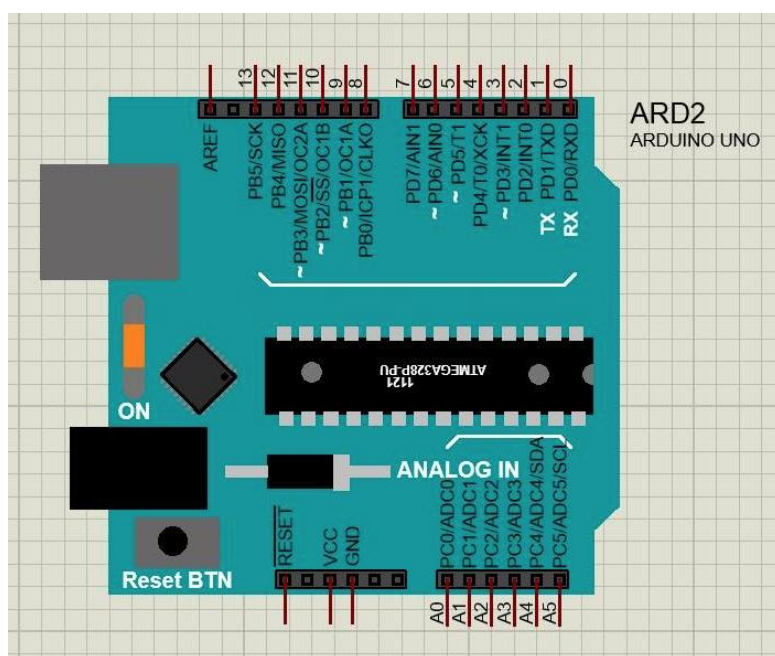
- یک موتور سروو که توسط گره لبه کنترل می شود تا گلدان را بچرخاند.

بخش های تمرین

این تمرین در قالب ۴ وظیفه انجام میشود.

وظیفه ۱: طراحی سخت افزار

- مدار سامانه را در **Proteus** طراحی کنید.
- نرم افزارهای مورد نیاز: برای برنامه نویسی برای آردوینو نیاز به نصب برنامه ی **Arduino IDE** دارید که از این [لینک](#) قابل دریافت است. برای نصب **Proteus** می توانید از سایت های دانیود داخلی استفاده نمایید.
- برای استفاده از شبیه ساز **Arduino** در **Proteus** نیاز به نصب کتابخانه ی مربوط به آن را دارید. دانلود این کتابخانه و مراحل نصب آن را می توانید در این [لینک](#) پیگیری کنید. بعد از نصب این کتابخانه، می توانید همانند شکل ۲، بلاک مربوط را به **proteus** اضافه کنید.



شکل ۲. بلوک Arduino در Proteus

• گره های لبه:

○ اجزا:

- Arduino Uno.
- Soil Moisture Sensor آنالوگ یا دیجیتال
- موتور DC برای شیر آب
- موتور سروو (برای چرخش گلدان).

○ وظایف:

- خواندن داده‌های سنسور رطوبت خاک و ارسال به گره مرکزی.
- اجرای فرمان‌های گره مرکزی (مانند شروع آبیاری یا چرخاندن گلدان).

• گره مرکزی:

○ اجزا:

- Arduino Uno.
- Temperature Sensor آنالوگ
- دو Light Sensor آنالوگ

○ وظایف:

- پایش دما و شدت نور.
- مقایسه شدت نور از دو سنسور و ارسال فرمان برای چرخاندن گلدان‌ها به سمت منبع نور قوی‌تر.
- ارسال فرمان آبیاری به گره‌های لبه بر اساس داده‌های رطوبت و دما.
- نکته: تعیین نرخ زمانی خواندن اطلاعات از سنسورها و ارتباط بین گره مرکزی و گره‌های لبه بر عهده خودتان است.

وظیفه ۲: ارتباط

• پیاده‌سازی ارتباط I2C:

- گره مرکزی به عنوان **Master**.
- هر گره لبه به عنوان **Slave** با آدرس I2C منحصر به فرد.
- نکته: تعیین آدرس‌ها بر روی باس I2C بر عهده خودتان است. همچنین پروتکل مشخصی برای تمایز بین داده‌های موجود بر روی باس، یعنی دستور چرخش گلدان، دستور باز کردن شیر، و داده‌های سنسور وجود ندارد. برای این منظور باید خودتان پروتوکلی برای این کار پیشنهاد دهید. برای مثال یک ایده می‌تواند استفاده از پرچم‌های (flag) ویژه‌ای برای جداسازی انواع داده باشد.

وظیفه ۳: توسعه نرم‌افزار

برای Arduino کد بنویسید و آپلود کنید:

۱. گره‌های لبه:

- کد برای خواندن داده‌های سنسور رطوبت خاک و پاسخ به فرمان‌ها از طریق I2C.
- کنترل موتور برای آبیاری و چرخاندن گلدان.

۲. گره مرکزی:

- کد برای خواندن داده‌های سنسور دما و نور.
- مقایسه شدت نور از دو سنسور و ارسال فرمان برای چرخاندن گلدان‌ها.

- تصمیم‌گیری برای آبیاری بر اساس داده‌های رطوبت و دما.
- ارسال فرمان‌ها به گره‌های لبه از طریق I2C

وظیفه ۴: شبیه‌سازی در Proteus

- کل سامانه را در Proteus شبیه‌سازی کنید.
- از اجزای مجازی مانند Potentiometer برای شبیه‌سازی ورودی سنسورها (مانند رطوبت خاک، دما و شدت نور) استفاده کنید.
- از LED ها (طبق هر کدینگی که خودتان طراحی کرده اید) برای نمایش وضعیت شیر آب و چرخش گلدان استفاده کنید.
- از ترمینال مجازی برای دیباگ و نمایش داده‌ها استفاده کنید.

معیارهای ارزیابی

۱. کامل بودن:
 - تمام قابلیت‌ها (پایش رطوبت، پایش دما، چرخاندن گلدان‌ها به سمت نور و آبیاری) پیاده‌سازی شده باشند.
 - ارتباط I2C بین گره مرکزی و گره‌های لبه به درستی کار کند.
۲. دقت:
 - سامانه به درستی به شرایط محیطی متغیر پاسخ دهد.
۳. شبیه‌سازی:
 - شبیه‌سازی به صورت صحیح و کامل در Proteus اجرا شود.
۴. کیفیت کد:
 - کدها خوانا و با توضیحات کافی باشند.

بخش‌های اختیاری

۱. استفاده از سنسور دیجیتال برای شدت نور، مانند BH1750 یا TSL2561 از طریق I2C به جای سنسور آنالوگ
۲. ارتباط گره‌ها از طریق wifi به جای i2c
۳. افزودن ماژول ساعت واقعی (RTC) برای اجرای برنامه‌های آبیاری زمان‌بندی شده.

موارد تحویلی

۱. مهلت تحویل: 1404/02/06
۲. پروژه به صورت گروهی انجام می‌شود. (گروه بندی در سامانه ایلرن نیز انجام می‌شود و تحویل تمرین حضوری و به صورت گروهی خواهد بود)

۳. هر گروه می بایست کار را تقسیم کنند. همچنین از Git برای ساختن branch و تقسیم issue ها استفاده نمایند. (با استفاده از commit ها و تعیین issue ها میزان مشارکت هر نفر مشخص می شود). بعد از انجام این کار کدها را در یک repository به نام CPS_IOT_HW در اکانت های GitHub خود قرار دهید (به صورت private). همچنین در یک فایل README.md می توانید report و داکيومنت خود را کامل کنید و در کنار repository قرار دهید. در نهایت لینک این repository را در محل پاسخ تمرین قرار دهید. (از فرستادن فایل به صورت زیپ جدا خودداری نمایید). اکانت تی ای های این تمرین رو به Repo خودتون به عنوان Maintainer به پروژه اضافه کنید.
۴. برای پیاده سازی این تمرین از C++ استفاده کنید.
۵. در انتها Repo شامل چهار فولدر اصلی خواهد بود: Client, Embedded, Proteus.
۶. ساختار صحیح و تمیزی کد برنامه، بخشی از نمره ی این پروژه شما خواهد بود. بنابراین در طراحی ساختار برنامه دقت به خرج دهید.
۷. برای هر قسمت کد، گزارش دقیق و شفاف بنویسید. کدهای ضمیمه شده بدون گزارش مربوطه نمره ای نخواهند داشت.
۸. هدف این تمرین یادگیری شماست. لطفا تمرین را خودتان انجام دهید. در صورت مشاهده ی تشابه بین کدهای دو گروه، مطابقت سیاست درس با گروه متقلب و تقلب دهنده برخورد خواهد شد.
۹. سؤالات خود را تا حد ممکن در گروه درس مطرح کنید تا سایر دانشجویان نیز از پاسخ آن بهره مند شوند.